Improvement on the Method for Automatic Differential Analysis and Its Application to Two Lightweight Block Ciphers DESL and LBlock-s

Siwei Sun[1,2], Lei Hu[1,2](✉), Kexin Qiao[1,2], Xiaoshuang Ma[1,2],
Jinyong Shan[1,2], and Ling Song[1,2]

[1] State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
[2] Data Assurance and Communication Security Research Center, Chinese Academy of Sciences, Beijing 100093, China
{sunsiwei,hulei,qiaokexin,maxiaoshuang,shanjinyong,songling}@iie.ac.cn

Abstract. With the development of the ubiquitous computing and large-scale information processing systems, the demand for lightweight block ciphers which is suitable for resource constrained computing devices is increasing. Hence, the methodology for design and analysis of block ciphers is becoming more important. In this paper, we use the Mixed-Integer Linear Programming (MILP) based tools for automatic differential cryptanalysis in a clever way to find improved single-key and related-key differential characteristics for DESL (a lightweight variant of the well known Data Encryption Standard), and obtain tighter security bound for LBlock-s (a core component of an lightweight authenticated encryption algorithm submitted to the international competition for authenticated encryption – CAESAR) against related-key differential attack. To be more specific, in searching for improved characteristics, we restrict the differential patterns allowed in the first and last rounds of the characteristics in the feasible region of an MILP problem by imposing different constraints than other rounds, and we partition the differential patterns of the DESL S-box into different sets with 2-bit more information associated with each pattern according to their probabilities. In addition, we show how to use the Gurobi optimizer combined with a known good characteristic to speed up the characteristic searching and bound proving process. Using these techniques, we managed to find the currently known best 9-round related-key differential characteristic for DESL, and the first published nontrivial related-key and single-key differential characteristics covering 10 rounds of DESL. Also, we obtain the currently known tightest security bound for LBlock-s against related-key differential attack. These techniques should be useful in analysis and design of other lightweight block ciphers.

Keywords: Automatic cryptanalysis · Security evaluation · Related-key differential attack · Authenticated encryption · Mixed-Integer Linear Programming

K. Tanaka and Y. Suga (Eds.): IWSEC 2015, LNCS 9241, pp. 97–111, 2015.
DOI: 10.1007/978-3-319-22425-1_7

1 Introduction

Cryptography plays a central role in protecting today's information system, and block cipher is one of the most important types of cryptographic algorithms. Moreover, with the development of the ubiquitous computing and large-scale information processing systems, the demand for lightweight block ciphers which are suitable for resource constrained computing devices such as sensor nodes, and RFID tags is increasing. Therefore, design and analysis of lightweight block ciphers draw much attention from the researchers in applied cryptography.

Differential cryptanalysis [20], introduced by Eli Biham and Adi Shamir in the late 1980s, is one of the most effective attacks on modern block ciphers. Moreover, many cryptanalytic techniques, such as the related-key differential attack [2,6,19], truncated differential attack [24], statistical saturation attack [11,16], impossible differential attack [1,23], (probabilistic) higher order differential attack [24,43], boomerang attack [18], multiple differential attack [8,9,15,17], differential-linear cryptanalysis [42], multiple linear attack [5,14,28–30] and so on so forth, are essentially based on differential attack.

Typically, the first step in differential attack is to find a differential characteristic with high (or the highest possible) probability. Hence, a method used for searching for good (or the best) differential characteristic is of great importance. For a designer, such method can be used to obtain a proven security bound against differential attack, which is a necessary part of the design of a block cipher. In fact, a large part of the design document of a modern block cipher is devoted to the security evaluation of differential attack. For an attacker, such method can be used to find high probability differential characteristics of a cipher which lead to distinguishers or key recovery attacks.

Matusi's branch-and-bound depth-first search algorithm [31] is a classic method for finding the best differential characteristic of a cipher. Several works were devoted to improving the efficiency of Matsui's approach. The concept of *search pattern* was introduced in [34] to reduce the search complexity of Matusi's algorithm by detecting unnecessary search candidates. Further improvements were obtained by Aoki *et al.* [10] and Bao *et al.* [48]. Remarkably, significant efficiency improvement on Matsui's approach was observed for specific ciphers in [48].

Despite its guarantee for finding the best single-key differential characteristic for a cipher (given unlimited computational power), Matsui's algorithm and its variants has some important limitations making it not practically applicable in many situations. Firstly, for most ciphers, the original algorithm of Matsui is not practically applicable in the related-key model. Even though there exits Matsui's variant (see Biryukov *et al.*'s work [3]) for finding related-key differential characteristics, this method is not very useful for ciphers with nonlinear key schedule algorithms, whereas ciphers with nonlinear key schedule algorithms are plentiful. Moreover, with the developement of new techniques for cryptanalysis (*e.g.*, the differential fault attack [13,21,47] and biclique attack [7]), the related-key model is becoming more important and highly relevant to the design and analysis of symmetric-key cryptographic algorithms. Secondly, Matsui's approach is inefficient in finding the best characteristics for many ciphers, and some speeding-up techniques for Matusi's approach were intimately related to the special properties of the specific ciphers under consideration, making it difficult to implement and far from being a generic and convenient tool for cryptanalysis.

For ciphers that cannot be analyzed by Matsui's approach, the cryptanalysts turn to other methods which can be employed to find reasonably good characteristics. Although the characteristics found by these methods are not guaranteed to be the best, they do produce currently the best known results for many ciphers.

In [4], Biryukov *et al.*extend Matsui's algorithm by using the *partial* (rather than the full) difference distribution table (pDDT) to prevent the number of explored candidates from exploding and at the same time keep the total probability of the resulting characteristic high. In [35], truncated differentials with the minimum number of active S-boxes are found by a breadth-first search based on the Dijkstra's algorithm, and then these truncated differentials are instantiated with actual differences.

Another line of research is to model the differential behavior of a cipher as an SAT or Mixed-Integer Linear Programming (MILP) problem which can be solved automatically by SAT or MILP solvers. Compared with other methods, these methods are easier to implement and more flexible. In [32,40,41], SMT/SAT solvers are employed to find differential characteristics of Salsa and other ciphers. Mouha *et al.* [33], Wu *et al.* [36], and Sun *et al.* [37] translated the problem of counting the minimum number of differentially active S-boxes into an MILP problem which can be solved automatically with open source or commercially available optimizers. This method has been applied in evaluating the security against (related-key) differential attacks of many symmetric-key schemes. However, this tool cannot be used to find the actual differential characteristics directly. In Asiacrypt 2014, two systematic methods for generating linear inequalities describing the differential properties of an arbitrary S-box were given in [39]. With these inequalities, the authors of [39] were able to construct an MILP model whose feasible region is a more accurate description of the differential behavior of a given cipher. Based on such MILP models, the authors of [39] proposed a heuristic algorithm for finding actual (related-key) differential characteristics, which is applicable to a wide range of block ciphers. In [38], Sun *et al.* get rid of the heuristic argument in [39] by constructing MILP models whose feasible regions are exactly the sets of all (related-key) differential characteristics.

These MILP based methods [37,39] mainly focus on finding characteristics with the minimum (or reasonably small) number of active S-boxes. However, it is well possible that a characteristic with more active S-boxes is better than a characteristic with a smaller number of active S-boxes. Even though a method for finding the best characteristic of a cipher by encoding the probability information into the differential patterns is proposed in [38], this method is only applicable to ciphers with 4 × 4 S-boxes and infeasible when the number of rounds is large. Therefore, by using these methods, we may miss some better characteristics. In this paper, we mainly focus on how to use the MILP based methods in a clever way such that better characteristics can be found.

Our Contribution. Based on Sun *et al.*'s MILP framework for automatic differential analysis presented in [37–39], we propose several techniques which are useful for finding improved characteristics. To be more specific, we restrict the differential patterns allowed in the first and last rounds to be those with relatively high probability in the differential distribution table, which makes sure that the active S-boxes in the first and last rounds assume relatively high probabilities. We also partition the differential patterns into different sets. For each of these sets, we associate 2-bit more information into its differential patterns, and try to find a characteristic maximizing a special objective function rather than maximizing the number of differentially active S-boxes. Moreover, after we find a good characteristic with N_A active S-boxes, we use the tool presented in [38] to enumerate all characteristics with N_A, N_A+1 and N_A+2 active S-boxes, from which we may find better characteristics than the original one. We also present some tricks in using the Gurobi [22] optimizer which may speed up the solving process.

With these techniques, we find a related-key characteristic covering 9 rounds of DESL whose probability is $2^{-41.89}$, while the best previously published 9-round related-key differential characteristic with probability $2^{-44.06}$ is given in [38]. Note that in [3], only the upper bound of the probability of the related-key differential characteristics covering 9 rounds of DESL is given. We also present a 10-round single-key and related-key differential characteristics of DESL with probabilities $2^{-52.25}$ and $2^{-51.85}$ respectively. Moreover, we give so far the tightest security bound of the full LBlock-s with respect to related-key differential attack.

Organization of the Paper. In Sect. 2, we introduce the MILP framework for automatic differential cryptanalysis. In Sect. 3, we present several techniques for finding improved (related-key) differential characteristics. Then we apply these techniques to DESL and LBlock-s in Sect. 4. Section 5 is the conclusion and discussion.

2 MILP Based Framework for Automatic Differential Cryptanalysis

A brief introduction of Sun *et al.*'s method is given below. Sun *et al.*'s method [37–39] is an extension of Mouha *et al.*'s technique [33] based on Mixed-Integer Linear Programming, which can be used to search for (related-key) differential characteristics and obtain security bounds of a cipher with respect to the (related-key) differential attack automatically.

Sun *et al.*'s method is applicable to ciphers involving the following three operations:

- bitwise XOR;
- bitwise permutation L which permutes the bit positions of an n dimensional vector in $\mathbb{F}_2^n$;
- S-box, $\mathcal{S} : \mathbb{F}_2^\omega \rightarrow \mathbb{F}_2^\nu$.

Note that a general linear transformation $T : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ can be treated as some XOR summations and bitwise permutations of the input bits. In Sun *et al.*'s methods, a new variable x_i is introduced for every input and output bit-level differences, where $x_i = 1$ means the XOR difference at this position is 1 and $x_i = 0$ if there is no difference.

Also, for every S-box involved in the cipher, introduce a new 0–1 variable A_j such that

$$A_j = \begin{cases} 1, & \text{if the input word of the Sbox is nonzero,} \\ 0, & \text{otherwise.} \end{cases}$$

Now, we are ready to describe Sun *et al.*'s method by clarifying the objective function and constraints in the MILP model. Note that we assume that all variables involved are 0–1 variables.

Objective Function. The objective function is to minimize the sum of all variables A_j indicating the activities of the S-boxes: $\sum_j A_j$.

Constraints. Firstly, for every XOR operation $a \oplus b = c \in \{0, 1\}$, include the following constraints

$$\begin{cases} a + b + c \geq 2d_{\oplus} \\ a + b + c \leq 2 \\ d_{\oplus} \geq a, \; d_{\oplus} \geq b, \; d_{\oplus} \geq c \end{cases} \tag{1}$$

where $d_{\oplus}$ is a dummy variable.

Assuming $(x_{i_0}, \ldots, x_{i_{\omega-1}})$ and $(y_{i_0}, \ldots, y_{i_{\nu-1}})$ are the input and output differences of an $\omega \times \nu$ S-box marked by A_t, we have

$$\begin{cases} A_t - x_{i_k} \geq 0, \; k \in \{0, \ldots, \omega - 1\} \\ -A_t + \sum\limits_{j=0}^{\omega-1} x_{i_j} \geq 0 \end{cases} \tag{2}$$

and

$$\begin{cases} \sum\limits_{k=0}^{\omega-1} x_{i_k} + \sum\limits_{k=0}^{\nu-1} y_{j_k} \geq \mathcal{B}_{\mathcal{S}} d_{\mathcal{S}} \\ d_{\mathcal{S}} \geq x_{i_k}, \quad 0 \leq k \leq \omega - 1 \\ d_{\mathcal{S}} \geq y_{j_k}, \quad 0 \leq k \leq \nu - 1 \end{cases} \tag{3}$$

where $d_{\mathcal{S}}$ is a dummy variable, and the branch number $\mathcal{B}_{\mathcal{S}}$ of an S-box $\mathcal{S}$, is defined as $\min_{a \neq b}\{\text{wt}((a \oplus b) || (\mathcal{S}(a) \oplus \mathcal{S}(b)) : a, b \in \mathbb{F}_2^{\omega}\}$. For an bijective S-box we have

$$\begin{cases} \omega \sum\limits_{k=0}^{\nu-1} y_{j_k} - \sum\limits_{k=0}^{\omega-1} x_{i_k} \geq 0 \\ \nu \sum\limits_{k=0}^{\omega-1} x_{i_k} - \sum\limits_{k=0}^{\nu-1} y_{j_k} \geq 0 \end{cases} \tag{4}$$

Then, treat every possible input-output differential pattern $(x_0, \ldots, x_{\omega-1}) \rightarrow (y_0, \ldots, y_{\nu-1})$ of an $\omega \times \nu$ S-box as an $(\omega + \nu)$-dimensional vector $(x_0, \ldots, x_{\omega-1}, y_0, \ldots, y_{\nu-1}) \in \{0, 1\}^{\omega+\nu} \subseteq \mathbb{R}^{\omega+\nu}$, and compute the H-representation of the convex hull of all possible input-output differential patterns of the S-box. From the H-representation select a small number of linear inequalities using the greedy algorithm presented in [38] which can be used to exactly describe the differential behavior of the S-box. Finally, relate the input and output variables of the S-box using these inequalities. Now, if we require that all the variables involved are 0–1 variables, then the feasible region of the resulting MILP model is exactly the set of all differential characteristics. We mention here that all the constraints in (3) and (4) can be omitted if we have already use the constraints from the critical set, since these constraints remove all impossible patterns.

3 Techniques for Obtaining Better Characteristics

In [37–39], MILP models are constructed and solved to search for characteristics with a small or the minimal number of active S-boxes. The main reason preventing the solution of such MILP models from leading to better characteristics is that the objective function in the MILP model is to minimize the number of differentially or linearly active S-boxes. Under this setting, an MILP optimizer, say Gurobi, is constantly trying to find a characteristic with a smaller number of active S-boxes. In this process, some characteristics with higher probability but larger numbers of active S-boxes are lost. Therefore, the method presented in [37–39] may fail to find some better characteristics. In this section, we show how to mitigate this situation such that improved characteristics can be obtained automatically.

Technique 1. Finding Characteristics with More Active S-Boxes. For an iterative r-round block cipher E, Sun *et al.*'s methods can be used to find a characteristic with the minimal or a reasonably small number of active S-boxes. Assuming that such an r-round characteristic with N_A active S-boxes has been found, then we add the constraint $N_A \leq \sum_j S_j \leq N_A + m$ to the MILP model, where S_j's are the variables marking the activities of the involved S-boxes. Then we try to enumerate all related-key differential characteristics satisfying all the constraints in the model, where m is a small positive integer typically chosen to be 1 or 2. From these characteristics we may find better ones. This is in fact the same heuristic employed by Biham *et al.* in [12], where they try to find differential characteristics with higher probabilities for the amplified boomerang attack. They try to accomplish this by adding an active S-box in the first round. This might seem a bad thing (as this increase the number of active S-boxes), but they find out that in exchange they get 3 more differentials of the active S-boxes with probability 2^{-2} instead of 2^{-3}.

Technique 2. Imposing Different Constraints for Different Rounds. Another technique for getting better characteristic is to allow only those differential patterns in the first and last rounds of a characteristic to take relatively high probabilities. This is because the input of the first round and the output of the last round is relatively free when compared to other rounds. In fact, for every characteristic we get, we can always manually modify its input and output differences (such that high probability differential patterns are used in the first and last rounds) to get a characteristic which is at least not worse than the original one. This manual process can be done automatically in the MILP framework by, in the first and last rounds, using the constraints generated from the critical set of the convex hull of all differential patterns with probability higher than a threshold value p_T we choose, rather than the convex hull of all possible differential patterns. Note that it is important to do this automatically since the enumeration process may return thousands of characteristics.

Technique 3. Encoding More Information into the Differential Patterns of an S-Box. In Sect. 5 of [38], an MILP based method for constructing an MILP model which can be used to search for the best (related-key) differential characteristic of a block cipher with 4×4 S-box is proposed by encoding the probability information of the differentials of a 4×4 S-box into the differential patterns.

Take the PRESENT S-box S for example. For every possible differential pattern $(x_0, x_1, x_2, x_3) \rightarrow (y_0, y_1, y_2, y_3)$, a corresponding *differential pattern with probability information* can be constructed as follows

$$(x_0, x_1, x_2, x_3, y_0, y_1, y_2, y_3; p_0, p_1) \in \{0,1\}^{8+2}$$

where the two extra bits (p_0, p_1) are used to encode the differential probability $\mathrm{Pr}_S[(x_0, \ldots, x_{\omega-1}) \rightarrow (y_0, \ldots, y_{\nu-1})]$ as follows

$$\begin{cases} (p_0, p_1) = (0,0), \text{ if } \mathrm{Pr}_S[(x_0, \ldots, x_{\omega-1}) \rightarrow (y_0, \ldots, y_{\nu-1})] = 1; \\ (p_0, p_1) = (0,1), \text{ if } \mathrm{Pr}_S[(x_0, \ldots, x_{\omega-1}) \rightarrow (y_0, \ldots, y_{\nu-1})] = 2^{-2}; \\ (p_0, p_1) = (1,1), \text{ if } \mathrm{Pr}_S[(x_0, \ldots, x_{\omega-1}) \rightarrow (y_0, \ldots, y_{\nu-1})] = 2^{-3}. \end{cases} \quad (5)$$

Hence, the probability of the differential pattern $(x_0, x_1, x_2, x_3) \rightarrow (y_0, y_1, y_2, y_3)$ is $2^{-(p_0+2p_1)}$. We refer the reader to [38] for more information of the technique.

This technique is only feasible for ciphers for 4×4 S-boxes because there are only 3 different probabilities for all differential patterns of a typical 4×4 S-boxes and hence we need only $\lceil \log_2 3 \rceil = 2$ extra bits to encode the probability information for each differential pattern. For an $\omega \times \mu$ S-box, if d extra bits are needed to encode the differential probability information, then we need to compute the H-representation of the convex hull of a subset in $\mathbb{R}^{\omega+\mu+d}$. For the PRESENT S-box, we need to compute the H-representation of a convex hull of a subset in $\mathbb{R}^{4+4+2} = \mathbb{R}^{10}$. For the S-box of DESL, this technique is infeasible since there are 9 different probabilities for the differentials of the DESL S-box and we need at least $\lceil \log_2 9 \rceil = 4$ extra bits to encode the probability information. This will force us to compute the H-representation of a convex hull of a set in $\mathbb{R}^{6+4+4} = \mathbb{R}^{14}$ which leads to MILP models with too many constraints to be solved in practical time. In the following, we propose a technique which partitions the differential patterns into several sets according to their probabilities and encodes their probability information with less extra bits.

Definition 1. *Define $\mathcal{D}_S^{[p]}$ to be the set of all differential patterns of an $\omega \times \mu$ S-box with probability p, that is*

$$\mathcal{D}_S^{[p]} = \{(x_0, \cdots, x_{\omega-1}, y_0, \cdots, y_{\mu-1}) : \Pr_S[(x_0, \ldots, x_{\omega-1}) \to (y_0, \ldots, y_{\mu-1})] = p\},$$

and we use $\mathcal{D}_S^{[p_1, \cdots, p_t]}$ to denote the set $\mathcal{D}_S^{[p_1]} \cup \cdots \cup \mathcal{D}_S^{[p_t]}$.

Take the DESL S-box for example. For every possible differential pattern $(x_0, \cdots, x_5) \to (y_0, \cdots, y_3)$, we can construct a corresponding pattern

$$(x_0, \cdots, x_5, y_0, \cdots, y_3; \theta_0, \theta_1) \in \mathbb{R}^{6+4+2}$$

such that

$$\begin{cases} (\theta_0, \theta_1) = (0, 0), \text{ if } (x_0, \cdots, x_5, y_0, \cdots, y_3) \in \mathcal{D}_S^{[\frac{64}{64}]}; \\ (\theta_0, \theta_1) = (1, 0), \text{ if } (x_0, \cdots, x_5, y_0, \cdots, y_3) \in \mathcal{D}_S^{[\frac{16}{64}, \frac{14}{64}, \frac{12}{64}]}; \\ (\theta_0, \theta_1) = (0, 1), \text{ if } (x_0, \cdots, x_5, y_0, \cdots, y_3) \in \mathcal{D}_S^{[\frac{10}{64}, \frac{8}{64}, \frac{6}{64}]}; \\ (\theta_0, \theta_1) = (1, 1), \text{ if } (x_0, \cdots, x_5, y_0, \cdots, y_3) \in \mathcal{D}_S^{[\frac{4}{64}, \frac{2}{64}]}. \end{cases} \quad (6)$$

In this technique, the constraints for S-boxes are the critical sets of all patterns with the above encoding scheme, and the objective function is chosen to be minimizing $\sum(\theta_0 + \lambda\theta_1)$, where λ is a positive constant. Note that the differential patterns in $D^{[p_1, \cdots, p_t]}$ with larger p_i will lead to a smaller $\theta_0 + \lambda\theta_1$, and therefore tend to make the objective function $\sum(\theta_0 + \lambda\theta_1)$ smaller.

Note that this method is heuristic in nature. Firstly, unlike the case of PRESENT S-box, the encoding scheme does not represent the exact probability of a differential. Secondly, the solution which minimizes the objective function is not necessarily corresponding to the best characteristic. Finally, the partition of the differential patterns and the selection of λ are rather *ad-hoc* (we choose $\lambda = 3$ when applied this technique to DESL). All these problems deserve further investigation. In the next section, we will show that although this technique is heuristic and rather *ad-hoc*, it does produce the currently known best results for DESL.

Finally, we would also like to point out a feature provided by the MILP optimizer Gurobi [22] which may be useful in speeding up the searching process of better characteristics. In Gurobi, an MILP start (MST) file is used to specify an initial solution for a mixed integer programming model. The file lists values to assign to the variables in the model. If an MILP start file has been imported into an MILP model before optimization begins, the Gurobi optimizer will attempt to build a feasible solution from the specified start values. A good initial solution often speeds up the solution of the MILP model, since it provides an early bound on the optimal value, and also since the specified solution can be used to seed the local search heuristics employed by the MILP solver. An MILP start file consists of variable-value pairs, each on its own line. Any line that begins with the hash sign (#) is a comment line and is ignored. The following is a simple example:

```
# MIP start
x1  1
x2  0
x3  1
```

Therefore, by converting known good characteristics into an MST file, and importing it into our MILP model before optimization begins, we may speed up the searching process.

4 Application to DESL and LBlock-s

The techniques presented in Sect. 3 are implemented in a Python [44] framework, and we show its applications in the following.

4.1 Improved Single-Key and Related-Key Differential Characteristics for DESL

DESL [25] is a lightweight variant of the well known block cipher DES (the Data Encryption Standard), which is almost the same as DES except that it uses a single S-box instead of eight different S-boxes as in DES. This S-box has a special design criteria to discard high probability (single-key) differential characteristics. This simple modification makes DESL much stronger than DES with respect to differential attack. In [3], Alex Biryukov *et al.* observed that Matsui's tool is infeasible for finding the best differential characteristics for DESL. However, Matsui's tool can find the best characteristic of the full DES in no more than several hours on a PC. To the best of our knowledge, there is no published single-key or related-key differential characteristics covering 10 rounds of DESL, and in fact, even in the design document of DESL, there is no concrete security bounds provided for DESL.

We first generate an MILP model for 9-round DESL in the related-key model according to the technique 2 and technique 3 presented in Sect. 3. By solving this model using the Gurobi optimizer [22], we find a 9-round related-key differential characteristic for DESL with probability $2^{-41.89}$, which is the best published 9-round related-key differential characteristic so far. The concrete results are given in Tables 1 and 2.

Subsequently, we construct an MILP model for 10-round DESL in the related-key model. Before we start to solve this model, we import the 9-round related-key differential characteristic found previously as an MILP start file. Finally, we find a 10-round related-key differential characteristic of DESL with probability $2^{-51.85}$ and 14 active S-boxes (see Tables 3 and 4). Then by employing the technique 1 of Sect. 3, we search for characteristics with 14, 15 or 16 active S-boxes. Finally, we find a single-key (a special case of the related-key model where there is no key difference) differential characteristic with probability $2^{-52.25}$ and 15 active S-boxes, which is given in Table 5. Note that this is the first published nontrivial single-key differential characteristic covering 10 rounds of DESL.

Table 1. A 9-round related-key differential characteristic for DESL with probability $2^{-41.89}$ (characteristic in the encryption process)

Rounds	Left	Right
0	00000000000000010000000000000000	00000000010000000000000000000000
1	00000000010000000000000000000000	00000000000000000000000000000000
2	00000000000000000000000000000000	00000000010000000000000000000000
3	00000000010000000000000000000000	00000100000000000000000000000000
4	00000100000000000000000000000000	00000000010000000100000000000000
5	00000000010000000100000000000000	00100000000000000000000110000000
6	00100000000000000000000110000000	00000010110000000100000000000000
7	00000010110000000100000000000000	00000000000001000000000000000000
8	00000000000001000000000000000000	00000010100000000100000000000000
9	00000010100000000100000000000000	01100000000000000100000110010000

Table 2. A 9-round related-key differential characteristic for DESL with probability $2^{-41.89}$(characteristic in the key schedule algorithm)

Rounds	The differences in the key register
1	000000000000000010000000000000000000000000000000
2	000000000000000000000000000000000000000000000000
3	000000000000100000000000000000000000000000000000
4	000000000010000000000000000000000000000000000000
5	000000000000001000000000000000000000000000000000
6	010000000000000000000000000000000000000000000000
7	000000001000000000000000000000000000000000000000
8	000000000000000000000010000000000000000000000000
9	000000000000001000000000000000000000000000000000

Table 3. A 10-round related-key differential characteristic for DESL with probability $2^{-51.85}$ (characteristic in the key schedule algorithm)

Rounds	The differences in the key register
1	111111111111111111111111111111111111111111111101
2	111111111111111111111111111111111111111011111111
3	111111111111111111111111111111111111111111111111
4	111111111111111111111111101111111111111111111111
5	111111111111111111111111111111111111111111110111
6	111111111111111111111111111111111110111111111111
7	111111111111111111111111111111111111111111011111
8	111111111111111111111111111111111111011111111111
9	111111111111111111111111111111111111111111111111
10	111111111111111111111111011111111111111111111111

Table 4. A 10-round related-key differential characteristic for DESL with probability $2^{-51.85}$(characteristic in the encryption process)

Rounds	Left	Right
0	11111111111111111111111111011101	11011111111111111111111111111110
1	11011111111111111111111111111110	11111111111111111111111111011111
2	11111111111111111111111111011111	11011111111111111111111111111110
3	11011111111111111111111111111110	11111111111111010111111111111111
4	11111111111111010111111111111111	01011111111111111111111111111110
5	01011111111111111111111111111110	11111111111111010111010111111111
6	11111111111111010111010111111111	11111111111111111111111111111110
7	11111111111111111111111111111110	11111111111111111111111111011111
8	11111111111111111111111111011111	11111111111111111111111111111111
9	11111111111111111111111111111111	11111111111111111111111111011111
10	11111111111111111111111111011111	11011111111011111111101101111111

Table 5. A 10-round single-key differential characteristic for DESL with probability $2^{-52.25}$

Rounds	Left	Right
0	00000000000000010000000000000000	00000000010000000000000000000000
1	00000000010000000000000000000000	00000000000000000000000000000000
2	00000000000000000000000000000000	00000000010000000000000000000000
3	00000000010000000000000000000000	00000100000000000000000000000000
4	00000100000000000000000000000000	00000000010000000100000000000000
5	00000000010000000100000000000000	00100000000000000000000110000000
6	00100000000000000000000110000000	00000010110000000100000000000000
7	00000010110000000100000000000000	00000000000001000000000000000000
8	00000000000001000000000000000000	00000010100000000100000000000000
9	00000010100000000100000000000000	01100000000000000100000110010000

4.2 Tighter Security Bound for LBlock-s

LBlock is a lightweight block cipher proposed by Wu *et al.* in ACNS 2011 [46]. It is a Feistel Network with a 64-bit block size and a 80-bit key size. Since its publication, LBlock received extensive cryptanalysis, such as [27] and [45]. According to [45], the security of LBlock against biclique attack is not strong enough due to its relatively weak diffusion of the key schedule algorithm. So a new key schedule algorithm is proposed in [45]. The LBlock with this improved key schedule is called LBlock-s, which is a core component of the authenticated encryption LAC [26] submitted to the CAESAR competition (Competition for Authenticated Encryption: Security, Applicability, and Robustness). Also, instead of using 10 different S-boxes in LBlock, LBlock-s uses only one S-box to reduce the cost of hardware implementation. For a detailed description of the cipher LBlock-s we refer the reader to [45] and [26] for more information. In this section, we apply the technique presented in this paper and the method presented in [38,39] to LBlock-s, and we obtain so far the tightest security bound for the full LBlock-s.

To obtain the security bound for LBlock-s against related-key differential attack, we generate two MILP instances for 10-round and 11-round LBlock-s using the method presented in [38,39]. Then we use the Gurobi optimizer to solve these models. The results indicate that there are at least 10 active S-boxes for 10-round LBlock-s, and 11 active S-boxes for 11-round LBlock-s. However, when we solve the 11-round model, we use the 10-round related-key differential characteristic found by the Gurobi model by solving the 10-round model as an MILP start file (see Sect. 3), and import it into the 11-round model. Finally, we observe a roughly 7 % speed up compared with the case without using the MILP start file. Then, we employ the **technique 2** presented in Sect. 4.1 of [39] to generate an MILP model for 11-round LBlock-s such that only the differential patterns of the S-box with probability greater than or equal to 2^{-2} are allowed. By solving this model, we prove that there are at least 12 active S-boxes for 11-round LBlock-s in the related-key model if only those S-box differential patterns with probability greater than or equal to 2^{-2} are allowed. These results indicate that the probability of any related-key differential characteristic for the 11-round LBlock-s is at most $(2^{-2})^{10} \times 2^{-3} = 2^{-23}$. Consequently, the probability of the full round LBlock-s (32 rounds in total) is upper bounded by $2^{-23} \times 2^{-23} \times (2^{-2})^{10} = 2^{-66}$. Note that this is so far the tightest security bound for full LBlock-s against related-key differential attack.

5 Conclusion and Discussion

In this paper, we use the MILP based methods in a clever way to find better (related-key) differential characteristics of DESL and obtain tighter security bound for LBlock-s. The key idea is to force the active S-boxes in the first and last rounds of a characteristic to take the differentials with relatively high probabilities, encode more information into the differential patterns, and to find better characteristics by enumerating all characteristics with their objective value (number of active S-boxes) close to the minimum number of active S-boxes. Moreover, we show how to use Gurobi and a known good characteristic to speed up the searching process. Finally, we would like to propose some problems deserving further investigation. Firstly, how to find the related-key differential characteristic of DESL with the maximal probability automatically by using MILP technique? Secondly, how to exploit the special features and tune the available parameters of the Gurobi optimizer to speed up the solving process further?

Acknowledgements. The authors would like to thank the anonymous reviewers for their helpful comments and suggestions. The work of this paper was supported by the National Key Basic Research Program of China (2013CB834203), the National Natural Science Foundation of China (Grants 61472417, 61402469 and 61472415), the Strategic Priority Research Program of Chinese Academy of Sciences under Grant XDA06010702, and the State Key Laboratory of Information Security, Chinese Academy of Sciences.

References

1. Biryukov, A.: Impossible differential attack. In: van Tilborg, H.C.A., Jajodia, S. (eds.) Encyclopedia of Cryptography and Security, p. 597. Springer (2011)
2. Biryukov, A., Khovratovich, D.: Related-key cryptanalysis of the full AES-192 and AES-256. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 1–18. Springer, Heidelberg (2009)
3. Biryukov, A., Nikolić, I.: Search for Related-Key Differential Characteristics in DES-Like Ciphers. In: Joux, A. (ed.) FSE 2011. LNCS, vol. 6733, pp. 18–34. Springer, Heidelberg (2011)
4. Biryukov, A., Velichkov, V.: Automatic Search for Differential Trails in ARX Ciphers. In: Benaloh, J. (ed.) CT-RSA 2014. LNCS, vol. 8366, pp. 227–250. Springer, Heidelberg (2014)
5. Biryukov, A., De Cannière, C., Quisquater, M.: On multiple linear approximations. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 1–22. Springer, Heidelberg (2004)
6. Biryukov, A., Khovratovich, D., Nikolić, I.: Distinguisher and related-key attack on the full AES-256. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 231–249. Springer, Heidelberg (2009)
7. Bogdanov, A., Khovratovich, D., Rechberger, C.: Biclique Cryptanalysis of the Full AES. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 344–371. Springer, Heidelberg (2011)
8. Canteaut, A., Fuhr, T., Gilbert, H., Naya-Plasencia, M., Reinhard, J.-R.: Multiple differential cryptanalysis of round-reduced PRINCE. In: Cid, C., Rechberger, C. (eds.) FSE 2014. LNCS, vol. 8540, pp. 591–610. Springer, Heidelberg (2015)
9. Canteaut, A., Fuhr, T., Gilbert, H., Naya-Plasencia, M., Reinhard, J.-R.: Multiple differential cryptanalysis of round-reduced PRINCE (full version). IACR Cryptology ePrint Archive, Report 2014/089 (2014). http://eprint.iacr.org/2014/089
10. Aoki, K., Kobayashi, K., Moriai, S.: Best differential characteristic search of FEAL. In: Biham, E. (ed.) FSE 1997. LNCS, vol. 1267, pp. 41–53. Springer, Heidelberg (1997)
11. Collard, B., Standaert, F.-X.: A statistical saturation attack against the block cipher PRESENT. In: Fischlin, M. (ed.) CT-RSA 2009. LNCS, vol. 5473, pp. 195–210. Springer, Heidelberg (2009)
12. Biham, E., Dunkelman, O., Keller, N.: The rectangle attack - rectangling the serpent. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 340–357. Springer, Heidelberg (2001)
13. Boneh, D., DeMillo, R.A., Lipton, R.J.: On the importance of checking cryptographic protocols for faults. In: Fumy, W. (ed.) EUROCRYPT 1997. LNCS, vol. 1233, pp. 37–51. Springer, Heidelberg (1997)
14. Kaliski Jr, B.S., Robshaw, M.: Linear Cryptanalysis Using Multiple Approximations. In: Desmedt, Y.G. (ed.) CRYPTO 1994. LNCS, vol. 839, pp. 26–39. Springer, Heidelberg (1994)
15. Blondeau, C., Gérard, B.: Multiple differential cryptanalysis: theory and practice. In: Joux, A. (ed.) FSE 2011. LNCS, vol. 6733, pp. 35–54. Springer, Heidelberg (2011)
16. Blondeau, C., Nyberg, K.: Links between truncated differential and multidimensional linear properties of block ciphers and underlying attack complexities. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 165–182. Springer, Heidelberg (2014)

17. Blondeau, C., Gérard, B., Nyberg, K.: Multiple differential cryptanalysis using LLR and χ^2 statistics. In: Visconti, I., De Prisco, R. (eds.) SCN 2012. LNCS, vol. 7485, pp. 343–360. Springer, Heidelberg (2012)
18. Wagner, D.: The boomerang attack. In: Knudsen, L.R. (ed.) FSE 1999. LNCS, vol. 1636, pp. 156–170. Springer, Heidelberg (1999)
19. Biham, E.: New types of cryptanalytic attacks using related keys. J. Cryptol. **7**(4), 229–246 (1994)
20. Biham, E., Shamir, A.: Differential cryptanalysis of DES-like cryptosystems. J. Cryptol. **4**(1), 3–72 (1991)
21. Biham, E., Shamir, A.: Differential fault analysis of secret key cryptosystems. In: Kaliski Jr, B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 513–525. Springer, Heidelberg (1997)
22. Optimization, G.: Gurobi optimizer reference manual (2013). http://www.gurobi.com
23. Knudsen, L.: DEAL-a 128-bit block cipher. Complexity **258**(2), 216 (1998)
24. Knudsen, L.R.: Truncated and higher order differentials. In: Preneel, B. (ed.) FSE 1994. LNCS, vol. 1008, pp. 196–211. Springer, Heidelberg (1995)
25. Leander, G., Paar, C., Poschmann, A., Schramm, K.: New lightweight DES variants. In: Biryukov, A. (ed.) FSE 2007. LNCS, vol. 4593, pp. 196–210. Springer, Heidelberg (2007)
26. Zhang, L., Wu, W., Wang, Y., Wu, S., Zhang, J.: LAC: a lightweight authenticated encryption cipher. CAESAR submission (2014). http://competitions.cr.yp.to/round1/lacv1.pdf
27. Liu, Y., Gu, D., Liu, Z., Li, W.: Impossible differential attacks on reduced-round LBlock. In: Ryan, M.D., Smyth, B., Wang, G. (eds.) ISPEC 2012. LNCS, vol. 7232, pp. 97–108. Springer, Heidelberg (2012)
28. Hermelin, M., Nyberg, K.: Linear cryptanalysis using multiple linear approximations. IACR Cryptology ePrint Archive, Report 2011/93 (2011). https://eprint.iacr.org/2011/093
29. Hermelin, M., Cho, J.Y., Nyberg, K.: Multidimensional linear cryptanalysis of reduced round serpent. In: Mu, Y., Susilo, W., Seberry, J. (eds.) ACISP 2008. LNCS, vol. 5107, pp. 203–215. Springer, Heidelberg (2008)
30. Hermelin, M., Cho, J.Y., Nyberg, K.: Multidimensional extension of Matsui's algorithm 2. In: Dunkelman, O. (ed.) FSE 2009. LNCS, vol. 5665, pp. 209–227. Springer, Heidelberg (2009)
31. Matsui, M.: On correlation between the order of S-Boxes and the strength of DES. In: De Santis, A. (ed.) EUROCRYPT 1994. LNCS, vol. 950, pp. 366–375. Springer, Heidelberg (1995)
32. Mouha, N., Preneel, B.: Towards finding optimal differential characteristics for ARX: application to Salsa20. IACR Cryptology ePrint Archive, Report 2013/328 (2013). http://eprint.iacr.org/2013/328
33. Mouha, N., Wang, Q., Gu, D., Preneel, B.: Differential and linear cryptanalysis using mixed-integer linear programming. In: Wu, C.-K., Yung, M., Lin, D. (eds.) Inscrypt 2011. LNCS, vol. 7537, pp. 57–76. Springer, Heidelberg (2012)
34. Ohta, K., Moriai, S., Aoki, K.: Improving the search algorithm for the best linear expression. In: Coppersmith, D. (ed.) CRYPTO 1995. LNCS, vol. 963, pp. 157–170. Springer, Heidelberg (1995)
35. Fouque, P.-A., Jean, J., Peyrin, T.: Structural evaluation of AES and chosen-key distinguisher of 9-round AES-128. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part I. LNCS, vol. 8042, pp. 183–203. Springer, Heidelberg (2013)

36. Wu, S., Wang, M.: Security evaluation against differential cryptanalysis for block cipher structures. IACR Cryptology ePrint Archive, Report 2011/551 (2011). https://eprint.iacr.org/2011/551
37. Sun, S., Hu, L., Song, L., Xie, Y., Wang, P.: Automatic security evaluation of block ciphers with S-bP structures against related-key differential attacks. In: Lin, D., Xu, S., Yung, M. (eds.) Inscrypt 2013. LNCS, vol. 8567, pp. 39–51. Springer, Heidelberg (2014)
38. Sun, S., Hu, L., Wang, M., Wang, P., Qiao, K., Ma, X., Shi, ., Song, L., Fu, K.: Towards finding the best characteristics of some bit-oriented block ciphers and automatic enumeration of (related-key) differential and linear characteristics with predefined properties. Cryptology ePrint Archive, Report 2014/747 (2014). http://eprint.iacr.org/2014/747
39. Sun, S., Hu, L., Wang, P., Qiao, K., Ma, X., Song, L.: Automatic security evaluation and (related-key) differential characteristic search: application to SIMON, PRESENT, LBlock, DES(L) and other bit-oriented block ciphers. In: Sarkar, P., Iwata, T. (eds.) ASIACRYPT 2014. LNCS, vol. 8873, pp. 158–178. Springer, Heidelberg (2014)
40. Kölbl, S.: CryptoSMT - an easy to use tool for cryptanalysis of symmetric primitives likes block ciphers or hash functions. https://github.com/kste/cryptosmt
41. Kölbl, S., Leander, G., Tiessen, T.: Observations on the SIMON block cipher family. Cryptology ePrint Archive, Report 2015/145 (2015). http://eprint.iacr.org/2015/145
42. Langford, S.K., Hellman, M.E.: Differential-linear cryptanalysis. In: Desmedt, Y.G. (ed.) CRYPTO 1994. LNCS, vol. 839, pp. 17–25. Springer, Heidelberg (1994)
43. Iwata, T., Kurosawa, K.: Probabilistic higher order differential attack and higher order bent functions. In: Lam, K.-Y., Okamoto, E., Xing, C. (eds.) ASIACRYPT 1999. LNCS, vol. 1716, pp. 62–74. Springer, Heidelberg (1999)
44. Van Rossum, G., et al.: Python programming language. In: USENIX Annual Technical Conference (2007)
45. Wang, Y., Wu, W., Yu, X., Zhang, L.: Security on LBlock against biclique cryptanalysis. In: Lee, D.H., Yung, M. (eds.) WISA 2012. LNCS, vol. 7690, pp. 1–14. Springer, Heidelberg (2012)
46. Wu, W., Zhang, L.: LBlock: a lightweight block cipher. In: Lopez, J., Tsudik, G. (eds.) ACNS 2011. LNCS, vol. 6715, pp. 327–344. Springer, Heidelberg (2011)
47. Zhao, X.J., Wang, T., Guo, S.Z.: Fault-propagation pattern based dfa on spn structure block ciphers using bitwise permutation, with application to PRESENT and printcipher. Technical report, IACR Cryptology ePrint Archive, Report 2011/086 (2011)
48. Bao, Z., Zhang, W., Lin, D.: Speeding Up the search algorithm for the best differential and best linear trails. In: Lin, D., Yung, M., Zhou, J. (eds.) Inscrypt 2014. LNCS, vol. 8957, pp. 259–285. Springer, Heidelberg (2015)